MATHEMATICS OF COMPUTATION
Volume 77, Number 264, October 2008, Pages 2345–2373
S 0025-5718(08)02009-7
Article electronically published on May 1, 2008

THE CONSTRUCTION OF GOOD EXTENSIBLE RANK-1 LATTICES

JOSEF DICK, FRIEDRICH PILLICHSHAMMER, AND BENJAMIN J. WATERHOUSE

Abstract. It has been shown by Hickernell and Niederreiter that there exist generating vectors for integration lattices which yield small integration errors for $n = p, p^2, \dots$ for all integers $p \geq 2$. This paper provides algorithms for the construction of generating vectors which are finitely extensible for $n = p, p^2, \dots$ for all integers $p \geq 2$. The proofs which show that our algorithms yield good extensible rank-1 lattices are based on a sieve principle. Particularly fast algorithms are obtained by using the fast component-by-component construction of Nuyens and Cools. Analogous results are presented for generating vectors with small weighted star discrepancy.

1. Introduction

We are interested in approximating a high dimensional integral

$$I_s(f) = \int_{[0,1]^s} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$$

by an equal weight quadrature rule

$$Q_{n,s}(f) = \frac{1}{n} \sum_{k=0}^{n-1} f(\boldsymbol{t}_k),$$

where the quadrature points $\boldsymbol{t}_0, \dots, \boldsymbol{t}_{n-1}$ are deterministically chosen from $[0,1)^s$. Such rules are called quasi-Monte Carlo rules. After putting some restrictions on the integrand concerning its smoothness, the question of how to choose the quadrature points becomes important. Several branches of quasi-Monte Carlo rules, which are nowadays quite interwoven, are known. Those are lattice rules [21, 28], Kronecker type sequences [18, 19, 20] and digital nets and sequences; see [20, 21, 23] and the references therein.

In this paper we study lattice rules, where the quadrature points are given by $\boldsymbol{t}_k = \{k\boldsymbol{z}/n + \boldsymbol{\Delta}\}$, $k = 0, \dots, n-1$. Here $\{\boldsymbol{x}\} = \boldsymbol{x} - \lfloor \boldsymbol{x} \rfloor$ denotes the fractional part of each component of a real vector $\boldsymbol{x}$, $\boldsymbol{z}$ is an integer vector and $\boldsymbol{\Delta} \in [0,1)^s$. The shift

Received by the editor June 13, 2006 and, in revised form, December 8, 2006.

2000 *Mathematics Subject Classification.* Primary 11K45, 65C05, 65D30.

Key words and phrases. Quasi-Monte Carlo, numerical integration, extensible lattice rule, component-by-component construction.

This work was supported by the Austrian Research Foundation (FWF), Project S 9609, that is part of the Austrian Research Network "Analytic Combinatorics and Probabilistic Number Theory".

The support of the Australian Research Council under its Centre of Excellence program is gratefully acknowledged.

$\boldsymbol{\Delta}$ is either chosen $\mathbf{0}$ (for periodic functions) or i.i.d. (for non-periodic functions). We shall refer to the set of n such points as the point set $P_{n,s}(\boldsymbol{z}, \boldsymbol{\Delta}) = \{\boldsymbol{t}_k\}_{k=0}^{n-1}$ or $P_{n,s}(\boldsymbol{z})$ when $\boldsymbol{\Delta} = \mathbf{0}$ or when the shift is "averaged out" (see below). An arbitrary point set will be denoted by $P_{n,s}$. The quality of the lattice rule thus depends on the choice of the integer vector $\boldsymbol{z}$. Until now no explicit constructions of $\boldsymbol{z}$ were known (except for dimension $s = 2$), hence one had to resort to a computer search. Several construction algorithms for the generating vector $\boldsymbol{z}$ have been introduced and analysed [3, 16, 29, 31]. For certain reproducing kernel Hilbert spaces H_s of integrands one can obtain an explicit form of the root mean square worst-case error

$$e_{n,s}(P_{n,s}(\boldsymbol{z}); K_s) = \left(\int_{[0,1]^s} e_{n,s}^2(P_{n,s}(\boldsymbol{z}, \boldsymbol{\Delta}); K_s) \mathrm{d}\boldsymbol{\Delta} \right)^{1/2}, \tag{1}$$

where

$$e_{n,s}(P_{n,s}(\boldsymbol{z}, \boldsymbol{\Delta}); K_s) = \sup_{\substack{f \in H_s \\ \|f\| \le 1}} |\, I_s(f) - Q_{n,s}(f) \,|,$$

with K_s the reproducing kernel (see below). Such a formula can then be used for a computer search of the generating vector $\boldsymbol{z}$ of a lattice rule. The proofs of the upper bounds on the integration error for certain classes of integrands depend on some averaging argument where one obtains a bound on the average of the worst-case error over all (or a suitable subset of) generating vectors (note that we can restrict $\boldsymbol{z} \in \{1, \dots, n-1\}^s$). As there is always at least one generating vector which is better than average, one can introduce a suitable algorithm to find a generating vector better than average. This approach has been used successfully to show that search algorithms (Korobov algorithm [31] or component-by-component algorithm [3, 16, 29]) yield good generating vectors. In the case of component-by-component algorithms the generating vector can even be extended in the number of dimensions. Though a desirable property [10], an extension in the number of points on the other hand has until now not been shown to be possible with these types of algorithms.

That an extension of lattice rules in the number of points is possible at least theoretically has been shown in [11]. Such lattice rules are nowadays called extensible lattice rules. In this case the quadrature points are given by $\boldsymbol{t}_k = \{\varphi(k)\boldsymbol{z} + \boldsymbol{\Delta}\}$, where for $k = k_0 + k_1 p + \cdots + k_m p^m$ the radical inverse function φ is defined by $\varphi(k) = k_0 p^{-1} + \cdots + k_m p^{-m-1}$. Here $\boldsymbol{z}$ is a vector of p-adic numbers (see [13] for a definition of p-adic numbers; for our purposes here it is enough to assume that $\boldsymbol{z}$ is a vector of natural numbers, hence we will not introduce p-adic numbers here).

The existence of good extensible lattice rules has been proven in [13] (see Section 7 about a discussion of the precise meaning of "extensible"). Therein it was shown that there exists a generating vector $\boldsymbol{z}$ which yields a lattice rule which is good for all moduli $p, p^2, p^3, \dots$, for any integer $p \ge 2$. The proof is based on a more sophisticated averaging argument. It should be noted that the lattice rules whose existence was proven in [13] are extensible in both the number of points *and* the dimension. They share this property with lattice rules constructed by the CBC algorithm. After the existence had been established it remained a challenge to provide some construction algorithm which would yield generating vectors for extensible lattice rules. Several successful numerical investigations have been carried out [2, 12], but a proof that those algorithms yield good extensible lattice rules was not provided.

In this paper we provide such an algorithm together with a proof (see also [4] for an analogue for polynomial lattice rules). The argument for the proof is indeed similar to the one used in [13] (and also [22]). It uses a combination of Markov's inequality, Jensen's inequality and an extension of the following simple fact: let A, B be two subsets of a finite set N and let $|N|$ denote the number of elements in N. Then $|A|, |B| > |N|/2$ implies that $A \cap B \neq \emptyset$. (We use an extension of this to an arbitrary number of subsets of N.) Using these principles we can obtain both an algorithm and a proof, thereby first providing construction algorithms for extensible lattice rules. To speed up the algorithm we show that we can also use a component-by-component approach [29] together with the fast computation method introduced in [24, 25]. This way we obtain a practically feasible construction of extensible lattice rules in (for many applications) sufficiently large dimensions and range of moduli.

Unfortunately our construction algorithms are not extensible in both n and s simultaneously. The first sieve algorithm (see Section 3), though slow, is in principle extensible in n, but is not extensible in the dimension. The CBC sieve algorithm and the fast CBC sieve algorithm (see Section 4) construct generating vectors for a range of moduli and are extensible in the dimension, but once the vector is constructed it is not possible to extend the vector to also work well for other moduli. Hence the CBC sieve constructions provide embedded rather then extensible lattice rules [2]. Obtaining an algorithm which is extensible in both n and s simultaneously remains an interesting open question. See also Section 7 at the end of the paper for a discussion of this topic.

The rest of the paper is arranged as follows. In Section 2 we discuss the function space setting of weighted reproducing kernel Hilbert spaces. In Section 3 we introduce a sieve algorithm which outlines the ideas used for the algorithm and the proof. In Section 4 we extend the results from Section 3 by showing that the sieve algorithm can be combined with a component-by-component approach allowing us to efficiently construct generating vectors of high dimension. Further we also show that the fast CBC construction of [24] can be incorporated in the algorithm as well. Section 5 contains some brief numerical experiments, whereas in Section 6 we develop similar results to those in Section 3 which are based on minimising the quantity $R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$, rather than the worst-case error. Using these results we are able to construct extensible lattice point sets with small weighted star discrepancy. Finally, in Section 7 we state several remarks and mention some open problems.

2. Reproducing kernel Hilbert spaces

In this section we introduce classes of integrands for which we consider numerical integration. Reproducing kernel Hilbert spaces are nowadays widely used in numerical analysis and other areas and are also used here to define function classes of integrands. The theory of reproducing kernels was developed in [1]; see also [9] where reproducing kernel Hilbert spaces were used to investigate numerical integration.

A reproducing kernel Hilbert space over $[0,1]$ is a Hilbert space H admitting a function $K : [0,1] \times [0,1] \to \mathbb{R}$ such that $K(\cdot, y) \in H$ for all $y \in [0,1]$ and $\langle f, K(\cdot, y)\rangle_H = f(y)$ for all $y \in [0,1]$ and $f \in H$. A kernel function K with these properties is unique, and it can be shown that K is also symmetric and positive definite. For dimensions $s > 1$ we consider tensor products of one-dimensional spaces. It can be shown that the reproducing kernel for those spaces is just the product of the one-dimensional kernels, i.e., $K(\boldsymbol{x},\boldsymbol{y})=\prod_{j=1}^{s}K(x_j,y_j)$, where $\boldsymbol{x}=(x_1,\ldots,x_s)$ and $\boldsymbol{y}=(y_1,\ldots,y_s)$.

In the following we introduce the particular reproducing kernel Hilbert spaces in which numerical integration is frequently considered [3, 5, 6, 9, 13, 14, 16, 17, 28, 29, 30, 31].

2.1. **Weighted Sobolev spaces.** We consider a tensor product Sobolev space $H_{s,\boldsymbol{\gamma}}$ of absolutely continuous functions whose partial mixed derivatives of order one in each variable are square integrable. The norm in the unanchored weighted Sobolev space $H_{s,\boldsymbol{\gamma}}$ [5] is given by

$$\|f\|_{H_{s,\boldsymbol{\gamma}}}=\left(\sum_{u\subseteq\{1,\ldots,s\}}\prod_{j\in u}\gamma_j\int_{[0,1]^{|u|}}\left(\int_{[0,1]^{s-|u|}}\frac{\partial^{|u|}}{\partial\boldsymbol{x}_u}f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}_{\{1,\ldots,s\}\setminus u}\right)^2\mathrm{d}\boldsymbol{x}_u\right)^{1/2},$$

where $\partial^{|u|}/\partial\boldsymbol{x}_u f$ denotes the partial mixed derivative with respect to all variables $j\in u$. Here and in the rest of the paper the quantities γ_j are non-negative real numbers called weights which are introduced to modify the importance of different coordinate directions [30].

The reproducing kernel of the s-dimensional unanchored weighted Sobolev space [5] is given by

$$(2)\qquad K_{s,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y})=\prod_{j=1}^{s}\left(1+\gamma_j\left[\frac{1}{2}B_2(|x_j-y_j|)+\left(x_j-\frac{1}{2}\right)\left(y_j-\frac{1}{2}\right)\right]\right),$$

where $B_2(\cdot)$ denotes the Bernoulli polynomial of degree 2, given by

$$(3)\qquad B_2(x)=x^2-x+\frac{1}{6}=\frac{1}{2\pi^2}\sum_{h=-\infty}^{\infty}{}'\frac{\mathrm{e}^{2\pi\mathrm{i}hx}}{h^2}\quad\forall x\in[0,1].$$

Here and throughout this paper the notation $\sum'$ indicates a summation with the zero term excluded.

We can associate a shift-invariant kernel [9] with $K_{s,\boldsymbol{\gamma}}$ by setting

$$K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y})=\int_{[0,1]^s}K_{s,\boldsymbol{\gamma}}(\{\boldsymbol{x}+\boldsymbol{\Delta}\},\{\boldsymbol{y}+\boldsymbol{\Delta}\})\,\mathrm{d}\boldsymbol{\Delta}.$$

The shift-invariant kernel associated with $K_{s,\boldsymbol{\gamma}}$ is given by

$$(4)\qquad K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y})=\prod_{j=1}^{s}\left(1+\gamma_jB_2(|x_j-y_j|)\right).$$

Using these definitions it follows that the mean square worst-case error (1) for the weighted Sobolev space is given by [9]

$$(5)\qquad e^2_{n,s,\boldsymbol{\gamma}}(P_{n,s};K_{s,\boldsymbol{\gamma}})=\int_{[0,1]^{2s}}K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y})\,\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}-\frac{2}{n}\sum_{k=0}^{n-1}\int_{[0,1]^s}K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{t}_k)\,\mathrm{d}\boldsymbol{x}$$
$$+\frac{1}{n^2}\sum_{k=0}^{n-1}\sum_{i=0}^{n-1}K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{t}_i,\boldsymbol{t}_k),$$

which for a randomly shifted (extensible) lattice rule can be simplified to (see for example [29])

$$e^2_{n,s,\boldsymbol{\gamma}}(P_{n,s};K_{s,\boldsymbol{\gamma}}) = -1 + \frac{1}{n}\sum_{k=0}^{n-1} K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}(\boldsymbol{t}_k,\boldsymbol{0}) = -1 + \frac{1}{n}\sum_{k=0}^{n-1}\prod_{j=1}^{s}(1+\gamma_j B_2(t_{k,j})). \tag{6}$$

Note that the above formula can easily be evaluated using (3) for a given point set $P_{n,s}$.

Observe that the shift-invariant kernel $K^{\mathrm{sh}}_{s,\boldsymbol{\gamma}}$ is related to the reproducing kernel of a certain weighted Korobov space of periodic functions which we introduce in the following.

2.2. **Weighted Korobov spaces.** The s-dimensional weighted Korobov space $H_{\mathrm{per},s,\alpha,\boldsymbol{\gamma}}$ has a reproducing kernel of the form [9]

$$\begin{aligned} K_{\mathrm{per},s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y}) &= \prod_{j=1}^{s}\left(1+\gamma_j \sum_{h=-\infty}^{\infty}{}' \frac{\mathrm{e}^{2\pi\mathrm{i}h(x_j-y_j)}}{|h|^{\alpha}}\right) \\ &= \sum_{\boldsymbol{h}\in\mathbb{Z}^s} \frac{\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{h}\cdot(\boldsymbol{x}-\boldsymbol{y})}}{r_\alpha(\boldsymbol{h},\boldsymbol{\gamma})}, \end{aligned} \tag{7}$$

where for $\boldsymbol{h}=(h_1,\ldots,h_s)$,

$$r_\alpha(\boldsymbol{h},\boldsymbol{\gamma}) = \prod_{j=1}^{s} r_\alpha(h_j,\gamma_j) \quad \text{and} \quad r_\alpha(h_j,\gamma_j) = \begin{cases} 1 & \text{if } h_j = 0, \\ \gamma_j^{-1}|h_j|^{\alpha} & \text{if } h_j \neq 0. \end{cases}$$

The parameter α restricts the convergence of the Fourier coefficients of the functions in the Korobov space. Throughout the paper we will assume that $\alpha > 1$.

Equation (5) can again be used to obtain a formula for the worst-case error in the Korobov space $H_{\mathrm{per},s,\alpha,\boldsymbol{\gamma}}$,

$$e^2_{\mathrm{per},n,s,\alpha,\boldsymbol{\gamma}}(P_{n,s}(\boldsymbol{z});K_{\mathrm{per},s,\alpha,\boldsymbol{\gamma}}) = -1 + \frac{1}{n}\sum_{k=0}^{n-1}\prod_{j=1}^{s}\left(1+\gamma_j\sum_{h=-\infty}^{\infty}{}'\frac{\mathrm{e}^{2\pi\mathrm{i}khz_j/n}}{|h|^{\alpha}}\right) \tag{8}$$

$$= -1 + \frac{1}{n}\sum_{k=0}^{n-1}\sum_{\boldsymbol{h}\in\mathbb{Z}^s}\frac{\mathrm{e}^{2\pi\mathrm{i}k\boldsymbol{h}\cdot\boldsymbol{z}/n}}{r_\alpha(\boldsymbol{h},\boldsymbol{\gamma})} \tag{9}$$

$$= \sum_{\substack{\boldsymbol{h}\in\mathbb{Z}^s\setminus\{\boldsymbol{0}\} \\ \boldsymbol{h}\cdot\boldsymbol{z}\equiv 0 \pmod n}} \frac{1}{r_\alpha(\boldsymbol{h},\boldsymbol{\gamma})}. \tag{10}$$

It follows from (3), (6) and (8) that

$$e_{n,s,2\pi^2\boldsymbol{\gamma}}(P_{n,s}(\boldsymbol{z});K_{s,2\pi^2\boldsymbol{\gamma}}) = e_{\mathrm{per},n,s,2,\boldsymbol{\gamma}}(P_{n,s}(\boldsymbol{z});K_{\mathrm{per},s,\alpha,\boldsymbol{\gamma}}), \tag{11}$$

where $2\pi^2\boldsymbol{\gamma}$ denotes the sequence of weights $(2\pi^2\gamma_j)_{j\geq 1}$. Thus the results shown in the following are valid for the root mean square error for numerical integration in the Sobolev space as well as for the worst-case error for numerical integration in the Korobov space. Hence it is enough to state them only for $e_{\mathrm{per},n,s,\alpha,\boldsymbol{\gamma}}$ (equation (11) can also be used to obtain results for $e_{n,s,\boldsymbol{\gamma}}$).

In the following section we introduce the arguments used for obtaining an algorithm and a proof for the construction of good extensible lattice rules.

3. The sieve algorithm

In [13] the authors used p-adic numbers to show the existence of good extensible lattices. Basically we could use p-adic numbers, too, but as we focus on the construction of extensible lattices by computer search, which in practice (though theoretically possible if one lets the computer search infinitely long) can only be finite, it is enough in our case to assume that the generating vector is in the set $\mathbb{N}^s$. Throughout the paper let p be an arbitrary but fixed integer. Then we restrict the set of admissible generating vectors to

$$\mathcal{Z}_p^s = \{\boldsymbol{z} = (z_1, \ldots, z_s) \in \mathbb{N}^s : \gcd(z_j, p) = 1, j = 1, \ldots, s\}. \tag{12}$$

Clearly, there is an infinite number of elements in this set. Since $e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\widehat{\boldsymbol{z}}) = e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\overline{\boldsymbol{z}})$ if $\overline{\boldsymbol{z}} \equiv \widehat{\boldsymbol{z}} \pmod{p^m}$, we exploit the structure inherent in lattice rules by defining the set

$$\mathcal{Z}_{p,m}^s = \{\boldsymbol{z} \in \mathcal{Z}_p^s : z_j < p^m, j = 1, \ldots, s\}. \tag{13}$$

Where $s = 1$ for the above definitions, we will typically omit the superscript. For any positive integer m, a vector $\boldsymbol{z} \in \mathcal{Z}_p^s$ has some corresponding vector $\overline{\boldsymbol{z}} \in \mathcal{Z}_{p,m}^s$ such that $\boldsymbol{z} \equiv \overline{\boldsymbol{z}} \pmod{p^m}$. Note there are $\phi(p^m)^s$ elements in the set $\mathcal{Z}_{p,m}^s$, where ϕ is Euler's totient function.

3.1. Bounds on the worst-case error.

In this section we prove some essential results which will shed light on how we intend to construct good extensible lattice rules. In the following let $\zeta(\alpha) = \sum_{i=1}^\infty i^{-\alpha}$ denote the Riemann zeta function.

Theorem 1. *Let p, m and s be positive integers. Then we have*

$$\frac{1}{\phi(p^m)^s} \sum_{\boldsymbol{z} \in \mathcal{Z}_{p,m}^s} e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \le E^2_{p^m,s,\alpha,\boldsymbol{\gamma}},$$

where

$$E^2_{n,s,\alpha,\boldsymbol{\gamma}} = \frac{1}{n}\left(\prod_{j=1}^s \left(1 + 2^{\kappa+1}\gamma_j \zeta(\alpha)\right) - 1\right)$$

and κ is the number of distinct prime factors of n.

Further there exists a vector $\overline{\boldsymbol{z}} \in \mathcal{Z}_{p,m}^s$ such that

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\overline{\boldsymbol{z}}) \le E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for any $\lambda \in (1/\alpha, 1]$ where

$$E^2_{n,s,\alpha,\boldsymbol{\gamma}}(\lambda) = \frac{1}{n^{1/\lambda}}\left(\prod_{j=1}^s \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) - 1\right)^{1/\lambda}.$$

Proof. The first part of the theorem was proven in [17, Theorem 2.2]. Note that the actual theorem in [17] states that

$$E^2_{n,s,\alpha,\boldsymbol{\gamma}} = \frac{1}{n}\prod_{j=1}^s \left(1 + 2^{\kappa+1}\gamma_j \zeta(\alpha)\right);$$

however, it is clear from the workings of the proof in [17] that the theorem holds with the slightly improved bound of

$$E^2_{n,s,\alpha,\boldsymbol{\gamma}} = \frac{1}{n}\left(\prod_{j=1}^{s}\left(1+2^{\kappa+1}\gamma_j\zeta(\alpha)\right)-1\right)$$

as stated above.

The proof of the second part is similar to the proof of a result in [6] which makes use of Jensen's inequality, namely that for a sequence of positive numbers $\{a_k\}$

$$\sum_k a_k \le \left(\sum_k a_k^{\lambda}\right)^{1/\lambda} \quad \text{for all} \quad 0<\lambda\le 1. \tag{14}$$

Note that in the theorem we make the restriction $\lambda > 1/\alpha$ so that the function $\zeta(\alpha\lambda)$ is well defined. We will make this restriction throughout the paper.

Combining the worst-case error in (10) with Jensen's inequality we see that for any $\lambda \in (1/\alpha, 1]$ we have

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \le \left(e^2_{\mathrm{per},p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}(\boldsymbol{z})\right)^{1/\lambda}. \tag{15}$$

Here and in the following $\boldsymbol{\gamma}^{\lambda}$ denotes the sequence of weights $\boldsymbol{\gamma}^{\lambda} = (\gamma_1^{\lambda}, \gamma_2^{\lambda}, \ldots)$. From part 1 of this theorem, we know that

$$\frac{1}{\phi(p^m)^s}\sum_{\boldsymbol{z}\in\mathcal{Z}^s_{p,m}} e^2_{\mathrm{per},p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}(\boldsymbol{z}) \le E^2_{p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}},$$

and hence it follows that for any $\lambda \in (1/\alpha, 1]$ there exists a vector $\boldsymbol{z}_{\lambda} \in \mathcal{Z}^s_{p,m}$ such that $e^2_{\mathrm{per},p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}(\boldsymbol{z}_{\lambda}) \le E^2_{p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}$. Putting these two results together, we find that for any $\lambda \in (1/\alpha, 1]$ there exists a vector $\boldsymbol{z}_{\lambda} \in \mathcal{Z}^s_{p,m}$ such that

$$\left(e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_{\lambda})\right)^{\lambda} \le E^2_{p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}.$$

Now let $\overline{\boldsymbol{z}} \in \mathcal{Z}^s_{p,m}$ such that $e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\overline{\boldsymbol{z}}) = \min_{\boldsymbol{z}\in\mathcal{Z}^s_{p,m}} e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z})$. Then we obtain

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\overline{\boldsymbol{z}}) \le e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_{\lambda}) \le \left(E^2_{p^m,s,\alpha\lambda,\boldsymbol{\gamma}^{\lambda}}\right)^{1/\lambda} = E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for any $\lambda \in (1/\alpha, 1]$. This gives the desired result. □

We wish to define a probability measure over the set of all generating vectors $\mathcal{Z}^s_p$. We would like to do so such that the measure of corresponding vectors in $\mathcal{Z}^s_{p,m}$ is equiprobable. For $m \in \mathbb{N}$ let $\mu_{s,m}$ be the equiprobable measure on the set $\mathcal{Z}^s_{p,m}$. We say a subset A of $\mathcal{Z}^s_p$ is of finite type, if there exists an integer $m = m(A) \in \mathbb{N}$ and a subset A' of $\mathcal{Z}^s_{p,m}$ such that

$$A = \{\boldsymbol{z} \in \mathcal{Z}^s_p : (\boldsymbol{z} \pmod{p^m}) \in A'\}.$$

The measure of the finite type subset A is then defined as

$$\mu_s(A) = \mu_{s,m(A)}(A').$$

Thus,

$$\mu_s(A) = \frac{\#A'}{\phi(p^m)^s}. \tag{16}$$

(Of course, the number $m = m(A)$ is not uniquely defined by A since if m works, then also any number larger than m will work in the definition of a finite type subset. It is easy to see that (16) does not depend on the specific choice of m.)

We now define the following set. For a real $c \geq 1$ define the set

$$\mathcal{C}_{n,s,\alpha,\boldsymbol{\gamma}}(c) = \{\boldsymbol{z} \in \mathcal{Z}_p^s : e^2_{\mathrm{per},n,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq cE^2_{n,s,\alpha,\boldsymbol{\gamma}}\}. \tag{17}$$

This set has the following property.

Theorem 2. *Let p, m and s be positive integers. For any $c \geq 1$ we have*

$$\mu_s(\mathcal{C}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c)) > 1 - c^{-1}.$$

Proof. This follows immediately by applying Markov's inequality to the first part of Theorem 1. □

We now make a small adjustment to this set which allows us to incorporate Jensen's inequality; see [6] where a similar argument was used. For a real $c \geq 1$ define the set

$$\widetilde{\mathcal{C}}_{n,s,\alpha,\boldsymbol{\gamma}}(c) = \{\boldsymbol{z} \in \mathcal{Z}_p^s : e^2_{\mathrm{per},n,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq c^{1/\lambda}E^2_{n,s,\alpha,\boldsymbol{\gamma}}(\lambda) \text{ for all } 1/\alpha < \lambda \leq 1\}. \tag{18}$$

We obtain the following theorem.

Theorem 3. *Let p, m and s be positive integers. For any $c \geq 1$ we have*

$$\mu_s(\widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c)) > 1 - c^{-1}.$$

Proof. Let $c \geq 1$ be given and choose $1/\alpha < \lambda^* \leq 1$ such that $c^{1/\lambda^*}E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda^*) \leq c^{1/\lambda}E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \leq 1$. From Theorem 2 we see that

$$\mu_s(\mathcal{C}_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}(c)) > 1 - c^{-1}. \tag{19}$$

Now, if $\boldsymbol{z} \in \mathcal{C}_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}(c)$, then

$$e^2_{\mathrm{per},p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}(\boldsymbol{z}) \leq cE^2_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}.$$

By (15) this implies that

$$\left(e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z})\right)^{\lambda^*} \leq cE^2_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}},$$

which can be re-written as

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq c^{1/\lambda^*}\left(E^2_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}\right)^{1/\lambda^*} = c^{1/\lambda^*}E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda^*),$$

which in turn implies that $\boldsymbol{z} \in \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c)$. This means that $\mathcal{C}_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}(c) \subseteq \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c)$. Using (19) as a lower bound, we find that

$$\mu_s(\widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c)) \geq \mu_s(\mathcal{C}_{p^m,s,\alpha\lambda^*,\boldsymbol{\gamma}^{\lambda^*}}(c)) > 1 - c^{-1}.$$ □

In the following we will use the above theorem to construct lattices for a range of moduli.

3.2. **The sieve principle.** We now want to construct lattice rules which work well for several choices of m. Let p^{m_1} be the lowest number of points and p^{m_2} the highest number of points in which we are interested, i.e., $m_1 \leq m_2$. Then for each $m = m_1, \ldots, m_2$ we can define a set $\widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)$ as in (18). In order to obtain a generating vector which works well for all choices of $m = m_1, \ldots, m_2$ we need to show that the intersection $\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)$ is not empty, or equivalently, has measure greater than 0. To this end choose $c_m \geq 1$ such that

$$\sum_{m=m_1}^{m_2} c_m^{-1} \leq 1; \tag{20}$$

then the measure of the intersection of the sets above can be shown to be strictly positive.

In the following we will write $\widetilde{\mathcal{C}}^c_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)$ to denote the complement of the set $\widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)$ in $\mathcal{Z}_p^s$.

Theorem 4. *Let p and s be positive integers and let $0 < m_1 \leq m_2$. Let $c_m \geq 1$ for all $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \leq 1$. Then there exists a vector $\boldsymbol{z} \in \mathcal{Z}_p^s$ such that*

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \leq 1$ and $m = m_1, \ldots, m_2$.

Proof. We need to show that $\mu_s\left(\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)\right) > 0$. This is a simple calculation,

$$\begin{aligned}\mu_s\left(\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)\right) &= 1 - \mu_s\left(\bigcup_{m=m_1}^{m_2} \widetilde{\mathcal{C}}^c_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)\right)\\ &\geq 1 - \sum_{m=m_1}^{m_2} \mu_s(\widetilde{\mathcal{C}}^c_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m))\\ &> 1 - \sum_{m=m_1}^{m_2} c_m^{-1} \;\geq\; 0. \qquad \square\end{aligned}$$

The arguments used to prove Theorem 4 are very similar to the arguments used in [13]. (Using p-adic numbers we could indeed also allow m_2 to be infinite. As in [13], using the above arguments, it is also possible to show the existence of a large number of good generating vectors.) Perhaps an advantage of our presentation is that it is more apparent how an algorithm for the construction of good generating vectors can be obtained from the arguments in the proof. This is done in the following section.

3.3. **The sieve algorithm.** In this subsection we introduce the idea of how a good generating vector can be found by describing a sieve algorithm for the construction of a generating vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ which works well for $m = m_1, \ldots, m_2$. This algorithm is quite slow, but in later sections we will give some modifications which speed up the sieve algorithm.

We wish to find a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ which satisfies

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda) \quad \text{for all} \quad 1/\alpha < \lambda \leq 1 \text{ and } m = m_1, \ldots, m_2.$$

That is, we wish to find a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ that lies in $\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s,\alpha,\boldsymbol{\gamma}}(c_m)$. For $m = m_1$ we use a computer search to find $\lfloor (1 - c_{m_1}^{-1})\phi(p^{m_1})^s \rfloor + 1$ of the $\phi(p^{m_1})^s$ vectors in $\mathcal{Z}_{p,m_1}^s$, which satisfy $e^2_{\mathrm{per},p^{m_1},s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq c_{m_1}^{1/\lambda} E^2_{p^{m_1},s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \leq 1$, and we label this set T_{m_1}. By Theorem 3 we know that at least such a number of them exist.

We then construct the set S_{m_1+1} of all vectors $\boldsymbol{z} \in \mathcal{Z}_{p,m_1+1}^s$, such that there exists some $\overline{\boldsymbol{z}} \in T_{m_1}$ with $\boldsymbol{z} \equiv \overline{\boldsymbol{z}} \pmod{p^{m_1}}$. From the set S_{m_1+1} we only keep $\lfloor (1 - (c_{m_1}^{-1} + c_{m_1+1}^{-1}))\phi(p^{m_1+1})^s \rfloor + 1$ vectors which satisfy the inequality $e^2_{\mathrm{per},p^{m_1+1},s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq c_{m_1+1}^{1/\lambda} E^2_{p^{m_1+1},s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \leq 1$, and we label this set T_{m_1+1}.

Again by Theorem 3 we know there must be at least $\lfloor (1 - c_{m_1+1}^{-1})\phi(p^{m_1+1})^s \rfloor + 1$ vectors in $\mathcal{Z}_{p,m_1+1}^s$ which satisfy $e^2_{\mathrm{per},p^{m_1+1},s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq c_{m_1+1}^{1/\lambda} E^2_{p^{m_1+1},s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \leq 1$. Therefore, there must be at least $\lfloor (1 - (c_{m_1}^{-1} + c_{m_1+1}^{-1}))\phi(p^{m_1+1})^s \rfloor + 1$ vectors which satisfy $e^2_{\mathrm{per},p^{m_1},s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq c_{m_1}^{1/\lambda} E^2_{p^{m_1},s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \leq 1$ as well.

In the same way, we construct the sets $S_{m_1+2}, T_{m_1+2}, \ldots, S_{m_2}$ and T_{m_2}. Finally, by Theorem 4 above, T_{m_2} is guaranteed not to be empty. We may select $\boldsymbol{z}^*$ to be any vector from T_{m_2} (see Section 4.2 for some comments on how to choose a vector from T_{m_2}).

Remark 1. In principle we can allow m_2 to be infinite, i.e., we can choose c_m such that $\sum_{m=m_1}^{\infty} c_m^{-1} \leq 1$. Then we can stop the computer search at some finite $m' > m_1$. If one stores all the necessary values from the initial search, it is then also possible to resume the computer search at a later point in time to obtain an extensible lattice rule for moduli larger than $p^{m'}$. Hence the construction is truly extensible in the modulus (see also Section 7 for more information on extensibility). As we will show in the next section, the vector can also be extended in the dimension using a CBC approach, but once this is done, it becomes "embedded" (see [2]) rather than extensible in the modulus, since the values of m_1 and m_2 may not be altered once chosen.

Further, as can be seen from the arguments above, one need not choose successive values of m, i.e., one could choose an arbitrary subset $K \subset \mathbb{N}$ and construct a good lattice rule with p^m points for all $m \in K$. See also [4] for further comments.

Remark 2. The constants $c_m \geq 1$ for $m = m_1, \ldots, m_2$ may be chosen to be any positive sequence of reals such that (20) is satisfied. If m_2 is finite, one possible choice of c_m to satisfy (20) is $c_m = m_2 - m_1 + 1$. This corresponds to the lattice rule having in some sense the same quality for each value of m. This choice will be used later in Section 5.

If m_2 is chosen to be infinite, we cannot choose c_m to be independent of m as we did above. Instead, the constants c_m must grow with m sufficiently fast so that the sum in (20) converges. One possible choice is $c_m = Cm(\log(m+1))^{1+\epsilon}$ for any $\epsilon > 0$ where C is chosen to be larger than $\sum_{m=m_1}^{\infty} m^{-1}(\log(m+1))^{-(1+\epsilon)}$. This is the choice used in [13]. A similar choice would be $c_m = \zeta(1+\epsilon)m^{1+\epsilon}$ again for any $\epsilon > 0$.

The following theorem now applies to generating vectors constructed by the sieve algorithm.

Theorem 5. *Let p and s be positive integers and let $0 < m_1 \leq m_2$. Let $c_m \geq 1$ for all $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \leq 1$. Then the sieve algorithm constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that*

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \leq 1$ and $m = m_1, \ldots, m_2$.

Note that it is always possible to choose c_m of order $m^{1+\varepsilon}$ for some $\varepsilon > 0$; hence the factor c_m in the bound in the above theorem contributes at most one additional factor of $m^{(1+\varepsilon)/\lambda} = (\log n)^{(1+\varepsilon)/\lambda}$, where n is the number of points. It can be shown that for every $0 < \delta < 1$ there is a constant $D_\delta > 0$ such that $(\log n)^c n^{-1} \leq D_\delta n^{-\delta}$; hence for every $1/\alpha < \lambda \leq 1$ there is a constant $C_\lambda > 0$ such that

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq \frac{C_\lambda}{p^{m/\lambda}} \prod_{j=1}^{s} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right)^{1/\lambda}$$

for all $m = m_1, \ldots, m_2$. (Here the constant C_λ may depend on the particular choice of c_m; on the other hand there is also a constant C_λ even if $m_2 = \infty$; see [13].) So using the sieve algorithm we can construct generating vectors for lattice rules which achieve the optimal rate of convergence for a range of moduli.

4. The component-by-component sieve algorithm

In the previous section we gave the idea of how to construct extensible lattice rules. In this section we show that the sieve algorithm can be combined with a component-by-component (CBC) approach [29] to obtain a faster construction algorithm which will allow us to construct good lattice rules for a practically relevant range of moduli and dimensions. This also gives the added benefit of obtaining a construction which is also extensible in the dimension, but unfortunately the range of moduli in this case has to be chosen in advance and cannot be extended anymore. In this sense our lattice rules are embedded rather than extensible; see also Section 7 and [13].

4.1. **The CBC sieve algorithm.**

We may reduce the construction cost by constructing the vector $\boldsymbol{z}^*$ component-by-component. This approach has been shown to be very useful and effective in constructing lattice rules for fixed n where one has $\phi(n)^s$ choices of $\boldsymbol{z}$. In short, the CBC algorithm works in the following way: choose the first component of the generating vector $z_1^* = 1$. Then, for $\boldsymbol{z}_s^* = (z_1^*, \ldots, z_s^*)$ already chosen, we will choose a component z_{s+1}^* such that the worst-case error $e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*)$ satisfies a certain bound. This way we can obtain a good generating vector inductively [3, 16, 29].

We will now establish a similar sequence of theorems to those of Theorems 1–4 which now include the component-by-component approach. Since we construct a vector $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1})$ with $\boldsymbol{z}_s^*$ fixed, we are concerned only with the incremental impact of the choice of z_{s+1} on the worst-case error.

We shall require the following technical lemma.

Lemma 1. *For any positive integers p and m we have*

$$\sum_{k=0}^{p^m-1}\left|\frac{1}{\phi(p^m)}\sum_{z\in\mathcal{Z}_{p,m}}{\sum_{h=-\infty}^{\infty}}' \frac{\mathrm{e}^{2\pi \mathtt{i} khz/p^m}}{|h|^\alpha}\right| \le 2^{\kappa+1}\zeta(\alpha),$$

where κ is the number of distinct prime factors of p.

Proof. This follows directly from [17, Lemma 2.1 and Lemma 2.3]. □

We define the quantity

$$\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) = e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) - e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*)$$

which will be needed in the following.

We modify Theorem 1 as follows.

Theorem 6. *Let p, m and s be positive integers. Then we have*

$$\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}}\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \le \overline{\theta}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}$$

where

$$\overline{\theta}_{p^m,s+1,\alpha,\boldsymbol{\gamma}} = \frac{2^{\kappa+1}}{p^m}\gamma_{s+1}\zeta(\alpha)\prod_{j=1}^{s}(1+2\gamma_j\zeta(\alpha))$$

and κ is the number of distinct prime factors of p.

Proof. Note that by (10), $\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1})$ can be written in the form

$$\begin{aligned}\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) &= \sum_{\substack{(\boldsymbol{h},h_{s+1})\in\mathbb{Z}^{s+1}\setminus\{\boldsymbol{0}\}\\ (\boldsymbol{h},h_{s+1})\cdot(\boldsymbol{z}_s^*,z_{s+1})\equiv 0 \pmod{p^m}}}\frac{1}{r_\alpha((\boldsymbol{h},h_{s+1}),\boldsymbol{\gamma})}\\ &\quad - \sum_{\substack{\boldsymbol{h}\in\mathbb{Z}^s\setminus\{\boldsymbol{0}\}\\ \boldsymbol{h}\cdot\boldsymbol{z}_s^*\equiv 0 \pmod{p^m}}}\frac{1}{r_\alpha(\boldsymbol{h},\boldsymbol{\gamma})}\\ &= \sum_{\substack{(\boldsymbol{h},h_{s+1})\in\mathbb{Z}^{s+1},h_{s+1}\neq 0\\ (\boldsymbol{h},h_{s+1})\cdot(\boldsymbol{z}_s^*,z_{s+1})\equiv 0 \pmod{p^m}}}\frac{1}{r_\alpha((\boldsymbol{h},h_{s+1}),\boldsymbol{\gamma})}, \end{aligned} \tag{21}$$

and so each term $\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1})$ is non-negative. From (9) we see that

$$\begin{aligned}
&\frac{1}{\phi(p^m)} \sum_{z_{s+1}\in\mathcal{Z}_{p,m}} \theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \\
&= \left| \frac{1}{p^m} \sum_{k=0}^{p^m-1} \prod_{j=1}^{s} \left(1+\gamma_j \sideset{}{'}\sum_{h=-\infty}^{\infty} \frac{\mathrm{e}^{2\pi \mathrm{i} khz_j^*/p^m}}{|h|^\alpha}\right) \right. \\
&\qquad\qquad \left. \times \left[\frac{\gamma_{s+1}}{\phi(p^m)} \sum_{z_{s+1}\in\mathcal{Z}_{p,m}} \sideset{}{'}\sum_{h=-\infty}^{\infty} \frac{\mathrm{e}^{2\pi \mathrm{i} khz_{s+1}/p^m}}{|h|^\alpha} \right] \right| \\
&\le \frac{1}{p^m} \sum_{k=0}^{p^m-1} \prod_{j=1}^{s} \left| 1+2\gamma_j \sum_{h=1}^{\infty} \frac{\cos(2\pi khz_j^*/p^m)}{h^\alpha} \right| \\
&\qquad\qquad \times \left| \frac{\gamma_{s+1}}{\phi(p^m)} \sum_{z_{s+1}\in\mathcal{Z}_{p,m}} \sideset{}{'}\sum_{h=-\infty}^{\infty} \frac{\mathrm{e}^{2\pi \mathrm{i} khz_{s+1}/p^m}}{|h|^\alpha} \right| \\
&\le \frac{1}{p^m} \sum_{k=0}^{p^m-1} \prod_{j=1}^{s} (1+2\gamma_j\zeta(\alpha)) \left| \frac{\gamma_{s+1}}{\phi(p^m)} \sum_{z_{s+1}\in\mathcal{Z}_{p,m}} \sideset{}{'}\sum_{h=-\infty}^{\infty} \frac{\mathrm{e}^{2\pi \mathrm{i} khz_{s+1}/p^m}}{|h|^\alpha} \right| .
\end{aligned}$$

We see that the result now follows by Lemma 1. □

We now define a set which is analogous to (17). For a real $c \ge 1$ and $\boldsymbol{z}_s^* \in \mathcal{Z}_p^s$ let

$$(22)\qquad \mathcal{C}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*) = \{z_{s+1}\in\mathcal{Z}_p : \theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \le c\,\overline{\theta}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}\}.$$

The following theorem follows immediately from Markov's inequality. Recall that each term $\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1})$ is non-negative as seen in (21), and hence Markov's inequality can be applied. As in this section we only deal with sets of one-dimensional vectors; we simply write μ for the measure μ_1.

Theorem 7. *Let p, m and s be positive integers. Then for any $c \ge 1$ we have*

$$\mu(\mathcal{C}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c\,;\boldsymbol{z}_s^*)) > 1 - c^{-1}.$$

Proof. This follows immediately by applying Markov's inequality to Theorem 6. □

We will be able to achieve stronger convergence results for the worst-case error if we use Jensen's inequality. We define the set

$$\begin{aligned}
\widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*) \;=\; &\big\{z_{s+1}\in\mathcal{Z}_p : \theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \le c^{1/\lambda}\left(\overline{\theta}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}\right)^{1/\lambda} \\
(23)\qquad\qquad &\text{for all } 1/\alpha < \lambda \le 1\big\}.
\end{aligned}$$

This new set has the following property.

Theorem 8. *Let p, m and s be positive integers. Then for any $c \ge 1$ we have*

$$\mu(\widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*)) > 1 - c^{-1}.$$

Proof. From Theorem 7 we can say

$$(24)\qquad \mu(\mathcal{C}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(c;\boldsymbol{z}_s^*)) > 1 - c^{-1}.$$

Now, if $z_{s+1} \in \mathcal{C}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(c;\boldsymbol{z}_s^*)$, then

$$\theta_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(\boldsymbol{z}_s^*, z_{s+1}) \le c\,\overline{\theta}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}.$$

Applying Jensen's inequality to (21) we see that

$$(\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}))^\lambda \le \theta_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(\boldsymbol{z}_s^*, z_{s+1}).$$

Combining the last two inequalities implies

$$\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \le c^{1/\lambda} \left(\overline{\theta}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}\right)^{1/\lambda},$$

which implies that $z_{s+1} \in \widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*)$. This means that $\mathcal{C}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(c;\boldsymbol{z}_s^*) \subseteq \widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*)$, which by using (24) as a lower bound implies that

$$\mu(\widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c;\boldsymbol{z}_s^*)) \ge \mu(\mathcal{C}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}(c;\boldsymbol{z}_s^*)) > 1 - c^{-1}. \qquad \square$$

In the same vein as Theorem 4, we show in the following theorem that there exists a component $z_{s+1}^* \in \mathcal{Z}_p$ such that the worst-case error $e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*)$ is small for all $m = m_1, \ldots, m_2$.

Theorem 9. *Let p, m and s be positive integers. Let $\boldsymbol{z}_s^* \in \mathcal{Z}_p^s$. Let $c_m \ge 1$ for all $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \le 1$. Then there exists a $z_{s+1}^* \in \mathcal{Z}_p$ such that*

$$\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*) \le c_m^{1/\lambda} \left(\overline{\theta}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}\right)^{1/\lambda}$$

for all $1/\alpha < \lambda \le 1$ and $m = m_1, \ldots, m_2$.

Proof. We need to show that $\mu\left(\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c_m;\boldsymbol{z}_s^*)\right) > 0$. This is a simple calculation:

$$\begin{aligned}
\mu\left(\bigcap_{m=m_1}^{m_2} \widetilde{\mathcal{C}}_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(c_m;\boldsymbol{z}_s^*)\right) &= 1 - \mu\left(\bigcup_{m=m_1}^{m_2} \widetilde{\mathcal{C}}^c_{p^m,s+1}(c_m;\boldsymbol{z}_s^*)\right) \\
&\ge 1 - \sum_{m=m_1}^{m_2} \mu(\widetilde{\mathcal{C}}^c_{p^m,s+1}(c_m;\boldsymbol{z}_s^*)) \\
&> 1 - \sum_{m=m_1}^{m_2} c_m^{-1} \ \ge \ 0. \qquad \square
\end{aligned}$$

We can put the existing vector $\boldsymbol{z}_s^*$ together with the new component z_{s+1}^* to show that the vector $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1}^*)$ has the following properties.

Theorem 10. *Let p, m and s be positive integers. Let $\boldsymbol{z}_s^*$ be chosen so that*

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*) \le c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

and z_{s+1}^* be chosen so that

$$\theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*) \le c_m^{1/\lambda} \left(\overline{\theta}_{p^m,s+1,\alpha\lambda,\boldsymbol{\gamma}^\lambda}\right)^{1/\lambda}$$

for all $1/\alpha < \lambda \le 1$. Then

$$e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_{s+1}^*) \le c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \le 1$, where $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1}^*)$.

Proof. We have

$$
\begin{aligned}
&e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*) = e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*) + \theta_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*)\\
&\leq \frac{c_m^{1/\lambda}}{p^{m/\lambda}} \left(\prod_{j=1}^{s} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) - 1 \right)^{1/\lambda}\\
&+ \frac{c_m^{1/\lambda}}{p^{m/\lambda}} \left(2^{\kappa+1}\gamma_{s+1}^\lambda \zeta(\alpha\lambda) \prod_{j=1}^{s} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) \right)^{1/\lambda}\\
&\leq \frac{c_m^{1/\lambda}}{p^{m/\lambda}} \left(\prod_{j=1}^{s} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) - 1 + 2^{\kappa+1}\gamma_{s+1}^\lambda \zeta(\alpha\lambda) \prod_{j=1}^{s} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) \right)^{1/\lambda}\\
&= \frac{c_m^{1/\lambda}}{p^{m/\lambda}} \left(\prod_{j=1}^{s+1} \left(1 + 2^{\kappa+1}\gamma_j^\lambda \zeta(\alpha\lambda)\right) - 1 \right)^{1/\lambda} = c_m^{1/\lambda}\, E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda),
\end{aligned}
$$

where the second inequality uses another application of Jensen's inequality. □

We may now construct the extensible generating vector $\boldsymbol{z}^*$ using the CBC method. The algorithm to do this is stated formally in Algorithm 1.

Algorithm 1 CBC construction of $\boldsymbol{z}^*$ with small $e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$ for $m = m_1, \ldots, m_2$

```
Require: $m_1 \leq m_2 \in \mathbb{N}_0$, $\alpha > 1$, a positive sequence of weights $\boldsymbol{\gamma}$, $p$ and $s_{\max}$
    positive integers and a sequence $c_{m_1}, \ldots, c_{m_2}$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \leq 1$
Set $z_1^* = 1$
for $s = 1$ to $s_{\max} - 1$ do
    Find $\lfloor (1 - c_{m_1}^{-1})\phi(p^{m_1}) \rfloor + 1$ components $z_{s+1} \in \mathcal{Z}_{p,m_1}$ to populate the set
        $T_{m_1,s+1} \subseteq \{z_{s+1} \in \mathcal{Z}_{p,m_1} : e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \leq c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)$
            for all $1/\alpha < \lambda \leq 1\}$.
    for $m = m_1 + 1$ to $m_2$ do
        Define the set
            $S_{m,s+1} = \{z_{s+1} \in \mathcal{Z}_{p,m}, \exists\, \overline{z} \in T_{m-1,s+1}$ such that $z_{s+1} \equiv \overline{z} \pmod{p^{m-1}}\}$
        Find $\lfloor (1 - \sum_{i=m_1}^{m} c_i^{-1})\phi(p^m) \rfloor + 1$ vectors to populate the set
            $T_{m,s+1} \subseteq \{z_{s+1} \in S_{m,s+1} : e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \leq c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)$
                for all $1/\alpha < \lambda \leq 1\}$.
    end for
    Select $z_{s+1}^* \in T_{m_2,s+1}$
    Set $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1}^*)$
end for
Set $\boldsymbol{z}^* = \boldsymbol{z}_{s_{\max}}^*$
```

Theorem 11. *Let p and s be positive integers and $0 < m_1 \le m_2$. Let $c_m \ge 1$ for all $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \le 1$. Then Algorithm 1 constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that*

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \le c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \le 1$ and $m = m_1, \ldots, m_2$.

4.2. **Optimising the CBC sieve algorithm.** The classical CBC algorithm constructs one component of the generating vector at a time. For each dimension, it takes the component which minimises the worst-case error. The requirement that this component is the minimum is important in using Jensen's inequality to gain the optimal rate of convergence (see [16]). The sieve algorithm does not have this requirement. Rather than finding the minimiser at each step, we require a certain number of admissible vectors; that is, vectors whose worst-case error is lower than some bound. Therefore, Algorithm 1 will find an extensible lattice rule without the need for any optimisation.

However, it is instinctive that we should attempt to go beyond simply looking for a set of admissible vectors and attempting to find the *best* (in some sense) generating vectors at each step. This can be done by modifying the choice of the set $T_{m,s+1}$ for $m = m_1, \ldots, m_2$ and $s = 1, \ldots, s_{\max} - 1$ in Algorithm 1. Rather than just constructing $T_{m,s+1}$ with the first $\lfloor (1 - \sum_{i=m_1}^{m} c_i^{-1}) \phi(p^m) \rfloor + 1$ components z_{s+1} such that

$$e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \le c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \le 1$, we construct the set $T_{m,s+1}$ to contain *all* components that satisfy the bound. We then truncate the set $T_{m,s+1}$ to contain exactly those $\lfloor (1 - \sum_{i=m_1}^{m} c_i^{-1}) \phi(p^m) \rfloor + 1$ elements which have the smallest worst-case error $e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1})$, for the given $\boldsymbol{z}_s^*$. As we will see in the numerical section, the bound is significantly larger than the actual errors, and hence using such an optimisation step ensures that we do not choose a component which barely satisfies the bound but rather is amongst the best possibilities.

4.3. **The fast CBC sieve algorithm.** In the previous section we constructed the components of the generating vector by first choosing the first m_1 digits and then extending those up to m_2 digits step-by-step for a set of good components. Though this algorithm is feasible for practical values, it does not allow us to use the fast component-by-component algorithm introduced by Nuyens and Cools [24, 25]. Their construction algorithm reduces the usual construction cost of the CBC algorithm from $O(sn^2)$ to $O(sn \log n)$ (which is a remarkable speed-up for large n) by exploiting the structure of the calculation.

In order to make use of the fast CBC algorithm we modify the previous construction algorithms. In this case it is necessary to search over all possible choices of the new component z_{s+1}, rather than just those which have been shown to be *good* for earlier values of m. Here we simply store all the *good* components z_{s+1} for the generating vector $(\boldsymbol{z}_s^*, z_{s+1})$ for each value of m. The construction is then performed by minimising a new error measure, which, for given $\boldsymbol{z}_s^* \in \mathcal{Z}_{p,m_2}^s$, is defined by

$$F_{m_1,m_2,s+1,\alpha,\boldsymbol{\gamma}}(z_{s+1}) = \sum_{m=m_1}^{m_2} \max_{1/\alpha < \lambda \le 1} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)}. \tag{25}$$

Using this measure we now construct a generating vector component-by-component, in each step choosing z_{s+1}^* which minimises the quantity $F_{m_1,m_2,s+1,\alpha,\boldsymbol{\gamma}}$.

Algorithm 2 Fast CBC sieve construction of a good generating vector $\boldsymbol{z}^*$ with small $e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$

```
Require: $m_1 \le m_2 \in \mathbb{N}_0$, $\alpha > 1$, a positive sequence of weights $\boldsymbol{\gamma}$, $p$ and $s_{\max}$ positive integers and the positive sequence $c_{m_1}, \dots, c_{m_2}$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \le 1$

Set $z_1^* = 1$
for $s = 1$ to $s_{\max} - 1$ do
    for $m = m_1$ to $m_2$ do
        Compute $\lambda_m^* \in (1/\alpha, 1]$ which minimises $N_{m,c_m}(\lambda) = c_m^{1/\lambda} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda)$ as a function of $\lambda$.
        For each $z_{s+1,m} \in \mathcal{Z}_{p,m}$ compute
            $$\frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{N_{m,c_m}(\lambda_m^*)}.$$
    end for
    Set
        $$T_{s+1} = \left\{ z_{s+1} \in \mathcal{Z}_{p,m_2} : \max_{m_1 \le m \le m_2} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{N_{m,c_m}(\lambda_m^*)} \le 1 \right\}.$$
    Select $z_{s+1}^* \in T_{s+1}$ which minimises
        $$\sum_{m=m_1}^{m_2} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{N_{m,c_m}(\lambda_m^*)}.$$
    Set $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1}^*)$
end for
Set $\boldsymbol{z}^* = \boldsymbol{z}_{s_{\max}}^*$
```

We can now use Theorem 9 to show that there must be at least one choice of $z_{s+1} \in \mathcal{Z}_p$ which is good for all $m = m_1, \dots, m_2$.

Theorem 12. *Let p and s be positive integers and $0 < m_1 \le m_2$. Let $c_m \ge 1$ for all $m = m_1, \dots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \le 1$. Then Algorithm 2 constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that*

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \le c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \le 1$ and $m = m_1, \dots, m_2$.

Remark 3. In Algorithm 2 note that instead of choosing $z_{s+1}^* \in T_{s+1}$ which minimises the quantity

$$\sum_{m=m_1}^{m_2} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{N_{m,c_m}(\lambda_m^*)},$$

it would be enough in theory to pick any $z_{s+1}^* \in T_{s+1}$ (which must be non-empty by Theorems 9 and 10). But as searching for the minimum increases the construction cost only marginally, it is advisable to include this step since the bound is typically very loose (see Section 5). Note that another legitimate choice in line 8 would be to select the $z_{s+1}^* \in T_{s+1}$ which minimises

$$F'_{m_1,m_2,s+1,\alpha,\gamma}(z_{s+1}) = \max_{m_1 \le m \le m_2} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{N_{m,c_m}(\lambda_m^*)}. \tag{26}$$

In this instance we have chosen the former because it gives smaller worst case errors in the numerical experiments.

The minimum of $N_{m,c_m}(\lambda)$ can be found with sufficient accuracy using any standard one-dimensional constrained optimisation software. Computing the normalised worst-case error Algorithm 2 can be done in order $n \log n$ operations using the fast CBC algorithm of [25].

4.4. **Theoretical bounds on the algorithm of Cools et al.** Algorithm 2 is very similar in nature to the algorithm suggested in [2]. Their algorithm is different in that given $\boldsymbol{z}_s^*$ they choose z_{s+1}^* to minimise the error measure defined by

$$V_{p,m_1,m_2,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1})) = \max_{m_1 \le m \le m_2} \frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))}{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)})} \tag{27}$$

for $s = 1, \ldots, s_{\max} - 1$, where $\boldsymbol{z}^{(m)}$ is the generating vector with the CBC algorithm for $n = p^m$ for $m = m_1, \ldots, m_2$. In [2] there was no formal proof with any bound on the size of the error measure $V_{p,m_1,m_2,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z})$, although the numerical experiments suggested that it remained small.

Observe that the quality measures $F'_{m_1,m_2,s+1,\alpha,\gamma}(z_{s+1})$ (which is given by (26)) and $V_{p,m_1,m_2,s+1,\alpha,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1}))$ used in [2] are very similar. Indeed we will show in the following that with a few slight modifications we can change Algorithm 2 such that it is the same as the algorithm considered in [2].

Let $\lambda'_{m_1}, \ldots, \lambda'_{m_2} \in (1/\alpha, 1]$ (we will see later how those values could be chosen) and let

$$c_m = \left(\frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)})}{E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_m)} \right)^{\lambda'_m} \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{\lambda'_m \alpha}. \tag{28}$$

Further, choose λ_m^* in Algorithm 2 as λ'_m and select z_{s+1}^* in step 8 by minimising $F'_{m_1,m_2,s+1,\alpha,\gamma}(z_{s+1})$ with the constant c_m given by (28). Then

$$N_{m,c_m}(\lambda'_m) = c_m^{1/\lambda'_m} E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_m) = C e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)}),$$

where $C = \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{\alpha}$ is independent of m. This way we obtain the same algorithm as proposed by [2]. Note that the constant C does not have any influence on which z_{s+1}^* will be chosen and can actually be left out in Algorithm 2.

The basic principles used to obtain Theorem 12 now still apply as long as $\sum_{m=m_1}^{m_2} c_m^{-1} \leq 1$; that is,

$$\begin{aligned}
&\sum_{m=m_1}^{m_2} c_m^{-1} \\
&= \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{-\lambda'_m \alpha} \sum_{m=m_1}^{m_2} \left(\frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)})}{E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_m)} \right)^{-\lambda'_m} \\
&\leq \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{-1} \sum_{m=m_1}^{m_2} \left(\frac{e^2_{\mathrm{per},p^m,s+1,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)})}{E^2_{p^m,s+1,\alpha,\boldsymbol{\gamma}}(\lambda'_m)} \right)^{-\lambda'_m} \\
&= 1.
\end{aligned}$$

Hence we obtain the bound

$$e^2_{\mathrm{per},p^m,s_{\max},\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq e^2_{\mathrm{per},p^m,s_{\max},\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(m)}) \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s_{\max},\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s_{\max},\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{\alpha}$$

for all $m = m_1, \ldots, m_2$, where $\boldsymbol{z}^*$ is constructed by Algorithm 2 based on the quality measure $F'_{m_1,m_2,s+1,\alpha,\gamma}(z_{s+1})$ with the constant c_m given by (28). This shows that the error criteria used in [2] has to satisfy the bound

$$V_{p,m_1,m_2,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*_s) \leq \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{\alpha}$$

for $s = 1, \ldots, s_{\max}$.

The values $\lambda'_{m_1}, \ldots, \lambda'_{m_2}$ do not have any influence on the algorithm as seen above; they only appear in the bound above. Hence we have the following result.

Theorem 13. *Let p and $s_{\max}$ be positive integers and $0 < m_1 \leq m_2$. Let c_m be given by* (28) *for all $m = m_1, \ldots, m_2$. Then the modification of Algorithm* 2 *proposed above, or equivalently the construction algorithm used in* [2]*, constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that*

$$V_{p,m_1,m_2,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*_s) \leq \min_{1/\alpha < \lambda'_{m_1},\ldots,\lambda'_{m_2} \leq 1} \left(\sum_{k=m_1}^{m_2} \left(\frac{E^2_{p^k,s,\alpha,\boldsymbol{\gamma}}(\lambda'_k)}{e^2_{\mathrm{per},p^k,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})} \right)^{\lambda'_k} \right)^{\alpha}$$

for $s = 1, \ldots, s_{\max}$.

Compared with the numerical results in [2], the bound is certainly conservative. Further, the bound also depends on m_1 and m_2, as each summand in the sum in the bound above is at least 1.

Note that in the theory above we could also use the bound from [3, Theorem 6] instead of $E^2_{p^k,s,\alpha,\boldsymbol{\gamma}}(\lambda'_k)$, which states that the error is bounded by

$$e^2_{\mathrm{per},n,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*_s) \leq \frac{C^\alpha (\log\log n)^\alpha}{n^\alpha} \prod_{j=1}^{s} \left(1 + 2\gamma_j^{1/\alpha} (1 - \log 2 + \zeta(\alpha)^{1/\alpha} + \log n) \right)^{\alpha}$$

for all $n \in \mathbb{N}$ (we just used the fact that there is a constant C such that $\phi(n)^{-1} < C(\log\log n)/n$; see for example [8, Theorem 328]). If we use the lower bound from

TABLE 1. Worst-case error of the extensible lattice rule where $\gamma_j = 1/j^2$

	$e_{\mathrm{per},2^m,360,2,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$	$c_m^{1/2\lambda^*} E_{2^m,360,2,\boldsymbol{\gamma}}(\lambda^*)$	$U_{2,10,20,360,2,\boldsymbol{\gamma}}$
$m = 10$	8.20e-02	1.44e+00	5.71e-02
$m = 11$	5.33e-02	1.01e+00	5.25e-02
$m = 12$	3.41e-02	7.17e-01	4.76e-02
$m = 13$	2.21e-02	5.07e-01	4.37e-02
$m = 14$	1.44e-02	3.59e-01	4.00e-02
$m = 15$	9.41e-03	2.54e-01	3.71e-02
$m = 16$	5.81e-03	1.79e-01	3.24e-02
$m = 17$	3.73e-03	1.27e-01	2.94e-02
$m = 18$	2.37e-03	8.97e-02	2.65e-02
$m = 19$	1.53e-03	6.34e-02	2.41e-02
$m = 20$	9.89e-04	4.48e-02	2.20e-02

[26] instead of $e^2_{\mathrm{per},p^k,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{(k)})$, we obtain that the bound in Theorem 13 is at most of order m_2 to some power. Hence also for the algorithm of [2] the worst-case error for the extensible lattice rule can only be worse by a factor of m_2 to some power compared to the worst-case error for a lattice rule constructed by a component-by-component algorithm only for a fixed value of the number of points.

5. Numerical testing

We have shown that it is possible to construct a generating vector $\boldsymbol{z}^*$ such that

$$e^2_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \le c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$$

for all $1/\alpha < \lambda \le 1$, where $c_m \ge 1$ for $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} c_m^{-1} \le 1$. All the testing was performed using the fast CBC algorithm since it is the fastest computationally. There are several parameters for each calculation which we must choose. In each example we take $p = \alpha = 2$. We also assume that the constants c_m for $m = m_1, \ldots, m_2$ are equal for each m, that is, $c_m = m_2 - m_1 + 1$. For these experiments we take $m_1 = 10$, $m_2 = 20$ and $s_{\max} = 360$.

There are two conclusions which can be drawn from our numerical experiments. The first conclusion we may draw is that the worst-case error for the extensible lattice rule is much smaller than the bound in Theorem 12 suggests. To demonstrate this we define the quantity

$$U_{p,m_1,m_2,s_{\max},\alpha,\boldsymbol{\gamma}} = \max_{1 \le s \le s_{\max}} \frac{e_{\mathrm{per},p^m,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^{*(m_1,m_2)})}{c_m^{1/2\lambda^*} E_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda^*)}, \tag{29}$$

where $c_m^{1/\lambda^*} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda^*) \le c_m^{1/\lambda} E^2_{p^m,s,\alpha,\boldsymbol{\gamma}}(\lambda)$ for all $1/\alpha < \lambda \le 1$ and $\boldsymbol{z}^{*(m_1,m_2)}$ is an extensible lattice rule constructed with Algorithm 2. Theorem 12 shows that $U_{p,m_1,m_2,s_{\max},\alpha,\boldsymbol{\gamma}}$ is bounded by 1.

In Tables 1–3 we compare the values of $U_{2,10,20,360,2,\boldsymbol{\gamma}}$ for different choices of $\boldsymbol{\gamma}$. We see that in each case $U_{2,10,20,360,2,\boldsymbol{\gamma}}$ is an order of magnitude less than 1. In fact, the numerical tests do not find any examples where $U_{2,10,20,360,2,\boldsymbol{\gamma}}$ is greater than 0.062.

The second conclusion we may draw is that the worst-case error for the extensible lattice rule is not significantly greater than the worst-case error for the "near

TABLE 2. Worst-case error of the extensible lattice rule where $\gamma_j = 0.9^j$

	$e_{\text{per},2^m,360,2,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$	$c_m^{1/2\lambda^*} E_{2^m,360,2,\boldsymbol{\gamma}}(\lambda^*)$	$U_{2,10,20,360,2,\boldsymbol{\gamma}}$
$m = 10$	4.00e+02	3.50e+05	5.55e-02
$m = 11$	2.83e+02	2.47e+05	5.04e-02
$m = 12$	2.00e+02	1.75e+05	4.63e-02
$m = 13$	1.41e+02	1.24e+05	4.20e-02
$m = 14$	9.99e+01	8.75e+04	4.23e-02
$m = 15$	7.06e+01	6.18e+04	3.55e-02
$m = 16$	5.00e+01	4.37e+04	3.25e-02
$m = 17$	3.53e+01	3.09e+04	2.95e-02
$m = 18$	2.50e+01	2.19e+04	2.76e-02
$m = 19$	1.77e+01	1.55e+04	2.42e-02
$m = 20$	1.25e+01	1.09e+04	2.27e-02

TABLE 3. Worst-case error of the extensible lattice rule where $\gamma_j = 0.05$

	$e_{\text{per},2^m,360,2,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$	$c_m^{1/2\lambda^*} E_{2^m,360,2,\boldsymbol{\gamma}}(\lambda^*)$	$U_{2,10,20,360,2,\boldsymbol{\gamma}}$
$m = 10$	2.51e+10	1.80e+21	6.19e-02
$m = 11$	1.77e+10	1.27e+21	5.85e-02
$m = 12$	1.25e+10	9.01e+20	5.58e-02
$m = 13$	8.87e+09	6.37e+20	5.09e-02
$m = 14$	6.27e+09	4.51e+20	4.85e-02
$m = 15$	4.44e+09	3.19e+20	4.46e-02
$m = 16$	3.14e+09	2.25e+20	4.23e-02
$m = 17$	2.22e+09	1.59e+20	4.04e-02
$m = 18$	1.57e+09	1.13e+20	3.74e-02
$m = 19$	1.11e+09	7.96e+19	3.56e-02
$m = 20$	7.84e+08	5.63e+19	3.43e-02

optimal" lattice rule as constructed by the CBC algorithm. To demonstrate this we examine the error measure $V_{p,m_1,m_2,s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$ defined above.

In Figure 1 we see when $m_1 = 10$, $m_2 = 20$ and $p = \alpha = 2$, the greatest ratio of the worst-case error of the extensible lattice $\boldsymbol{z}^*$ and the worst-case error of the corresponding near optimal choice $\boldsymbol{z}^{(m)}$ (as constructed by the CBC algorithm) is always less than 2 for these particular choices of $\boldsymbol{\gamma}$. This is similar to the results in [2, Table 6.1].

6. EXTENSIBLE LATTICE RULES WITH SMALL STAR DISCREPANCY

In the sections above, we have developed three algorithms to construct an extensible lattice rule with small worst-case error. Another measure of the quality of a lattice rule (or any quasi-Monte Carlo rule for that matter) is the *weighted star discrepancy* of the underlying nodes. The weighted star discrepancy for a point set P_n consisting of n points in the s-dimensional unit cube is defined by

$$D^*_{n,s,\boldsymbol{\gamma}}(P_n) = \sup_{\emptyset \neq \mathfrak{u} \subseteq \{1,\ldots,s\}} \gamma_{\mathfrak{u}} \sup_{\boldsymbol{x}_{\mathfrak{u}} \in [0,1]^{|\mathfrak{u}|}} |\operatorname{disc}((\boldsymbol{x}_{\mathfrak{u}}, \mathbf{1}), P_n)|, \tag{30}$$

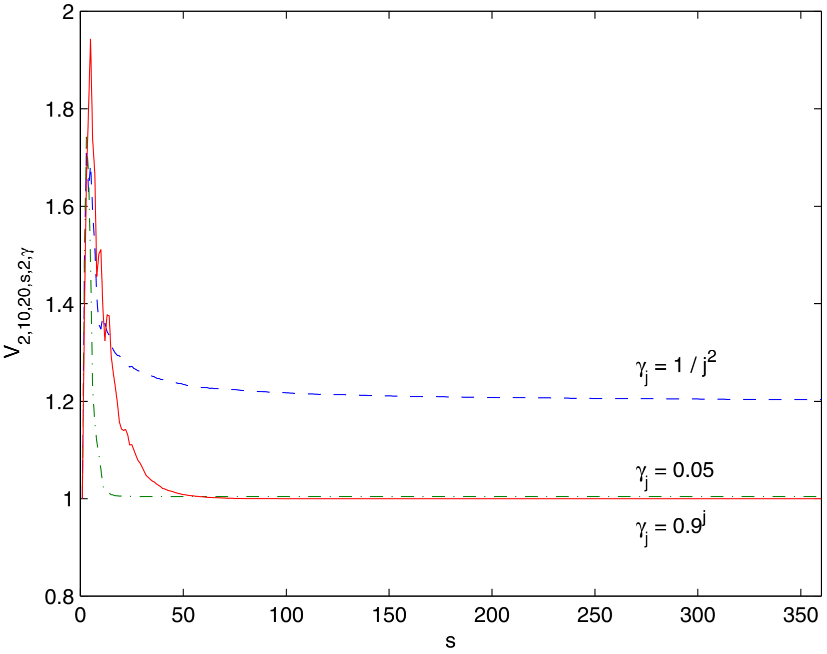

FIGURE 1. Graph of $V_{2,10,20,s,2,\boldsymbol{\gamma}}$ for 3 choices of $\boldsymbol{\gamma}$ and $s = 1, \ldots, 360$

where

$$\operatorname{disc}(\boldsymbol{x}, P_n) = \frac{\#(P_n \cap [\boldsymbol{0}, \boldsymbol{x}))}{n} - \operatorname{Vol}([\boldsymbol{0}, \boldsymbol{x})), \tag{31}$$

$\gamma_{\mathfrak{u}} = \prod_{j \in \mathfrak{u}} \gamma_j$ and for $\boldsymbol{x} = (x_1, \ldots, x_s)$ the vector $(\boldsymbol{x}_{\mathfrak{u}}, \boldsymbol{1})$ denotes the vector where the j-th component is x_j if $j \in \mathfrak{u}$ and 1 otherwise.

If P_n is the node set of a lattice rule with generating vector $\boldsymbol{z}$ we will in the following write $D^*_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$ instead of $D^*_{n,s,\boldsymbol{\gamma}}(P_n)$.

The quantity $D^*_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$ is difficult to compute. However, from [15] we easily deduce that the weighted star discrepancy is bounded by

$$D^*_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z}) < \frac{1}{2} R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z}) + \sum_{\emptyset \neq \mathfrak{u} \subseteq \{1,\ldots,s\}} \gamma_{\mathfrak{u}} \left(1 - \left(1 - \frac{1}{n}\right)^{|\mathfrak{u}|}\right),$$

where the quantity $R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$ is defined by

$$R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z}) = \sum_{\substack{\boldsymbol{h} \in B^*_{n,s} \\ \boldsymbol{h} \cdot \boldsymbol{z} \equiv 0 \pmod{n}}} \frac{1}{\widetilde{r}(\boldsymbol{h}, \boldsymbol{\gamma})}, \tag{32}$$

with

$$\widetilde{r}(\boldsymbol{h}, \boldsymbol{\gamma}) = \prod_{j=1}^{s} \widetilde{r}(h_j, \gamma_j) \quad \text{and} \quad \widetilde{r}(h_j, \gamma_j) = \begin{cases} (1 + \gamma_j)^{-1} & \text{if } h_j = 0, \\ \gamma_j^{-1} |h_j| & \text{if } h_j \neq 0, \end{cases}$$

and

$$B_{n,s} = \mathbb{Z}^s \cap (-n/2, n/2]^s \quad \text{and} \quad B^*_{n,s} = B_{n,s} \setminus \{\boldsymbol{0}\}.$$

In the case where $s = 1$, we will usually drop the subscript s.

It was proved by Joe [15] that if the vector of weights $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots)$ satisfies $\sum_{j=1}^{\infty} \gamma_j < \infty$, then we have

$$\sum_{\emptyset \neq \mathfrak{u} \subseteq \{1,\ldots,s\}} \gamma_{\mathfrak{u}} \left(1 - \left(1 - \frac{1}{n}\right)^{|\mathfrak{u}|}\right) \leq \frac{\max(1,\Gamma)\mathrm{e}^{\sum_{i=1}^{\infty}\gamma_i}}{n} \quad \forall s \geq 1,$$

where $\Gamma = \sum_{i=1}^{\infty} \frac{\gamma_i}{1+\gamma_i}$.

We therefore aim to construct extensible lattice rules with small weighted star discrepancy by minimising the quantity $R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$, for $n = p^m$ and $m = m_1, \ldots, m_2$.

In this section we will provide theorems analogous to theorems above which are based upon the worst-case error. These theorems can be used to derive algorithms which are almost identical to the ones above. Given the similarity between the theorems, some of the results below are stated without proof.

6.1. **The CBC sieve algorithm.** The following theorems are the natural analogs of Theorems 6–7 and Theorems 9–12.

The CBC sieve algorithm is Algorithm 1, where one replaces the worst-case error with the quantity $R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z})$ and $E_{p^m,s,\alpha,\boldsymbol{\gamma}}$ with $\overline{R}_{p^m,s,\boldsymbol{\gamma}}$, where

$$\overline{R}_{n,s,\boldsymbol{\gamma}} = \frac{1}{n}\left(\prod_{j=1}^{s}\left(1 + \gamma_j + \gamma_j\left(4\log n + \frac{2^{\kappa+1}\pi^2}{3}\right)\right) - \prod_{j=1}^{s}(1+\gamma_j)\right)$$

and κ is the number of distinct prime factors of n.

We will now prove that a CBC sieve algorithm similar to Algorithm 1 will construct a generating vector $\boldsymbol{z} \in \mathcal{Z}_{p,m}^s$ such that $R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}) \leq \overline{R}_{p^m,s,\boldsymbol{\gamma}}$. We begin by noting that when $s = 1$, and we take $z_1 = 1$, then the quantity $R_{p^m,1,\gamma_1}(z_1) = 0$ and is clearly less than $\overline{R}_{p^m,1,\gamma_1}$.

The quantity

$$\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) = R_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) - (1+\gamma_{s+1})R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*)$$

will be needed subsequently. It will also be more convenient to write $R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z})$ in the form

$$R_{n,s,\boldsymbol{\gamma}}(\boldsymbol{z}) = -\prod_{j=1}^{s}(1+\gamma_j) + \frac{1}{n}\sum_{k=0}^{n-1}\prod_{j=1}^{s}\left(1 + \gamma_j + \gamma_j \sum_{h_j \in B_n^*} \frac{\mathrm{e}^{2\pi\mathrm{i}kh_jz_j/n}}{|h|}\right), \tag{33}$$

which is easily derived by noting that

$$\frac{1}{n}\sum_{k=0}^{n-1}\mathrm{e}^{2\pi\mathrm{i}k\boldsymbol{h}\cdot\boldsymbol{z}/n} = \begin{cases} 1, & \text{if } \boldsymbol{h}\cdot\boldsymbol{z} \equiv 0 \pmod{n}; \\ 0, & \text{otherwise.} \end{cases}$$

The construction of $\boldsymbol{z}^*$ component-by-component is justified by the following theorem.

Theorem 14. *Let p, m and s be positive integers. Then*

$$\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}} \Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) \leq \overline{\Upsilon}_{p^m,s+1,\boldsymbol{\gamma}},$$

where

$$\overline{\Upsilon}_{n,s+1,\boldsymbol{\gamma}} = \frac{\gamma_{s+1}}{n}\left(4\log n + \frac{2^{\kappa+1}\pi^2}{3}\right)\prod_{j=1}^{s}\left(1+\gamma_j+\gamma_j\left(4\log n+\frac{2^{\kappa+1}\pi^2}{3}\right)\right).$$

Proof. Note first that using (33)

$$\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}) = \frac{\gamma_{s+1}}{p^m}\sum_{k=0}^{p^m-1}\left(C(k,p^m,z_{s+1})\prod_{j=1}^{s}(1+\gamma_j+\gamma_j C(k,p^m,z_j))\right),$$

where

$$C(k,n,z) = \sum_{h\in B_n^*}\frac{\mathrm{e}^{2\pi \mathtt{i} khz/n}}{|h|}.$$

Now, following a similar argument as in the proof of Theorem 6, and introducing the function $L(n) = \sum_{h\in B_n^*}\frac{1}{|h|}$, we see that

$$\begin{aligned}&\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}}\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1})\\ &\leq \frac{\gamma_{s+1}}{p^m}L(p^m)\prod_{j=1}^{s}(1+\gamma_j+\gamma_j L(p^m))\\ &\quad+\frac{\gamma_{s+1}}{p^m}\sum_{k=1}^{p^m-1}\left(\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}}C(k,p^m,z_{s+1})\right)\prod_{j=1}^{s}(1+\gamma_j+\gamma_j C(k,p^m,z_j)).\end{aligned}$$

While the quantity $L(n)$ may be computed exactly, for large n, it may be more practical to approximate $L(n)$ by the following function [21, Eq. (5.18)]:

$$L(n) = 2\log n + 2\overline{\gamma} - \log 4 + \epsilon(n) \quad \text{with} \quad |\epsilon(n)| < 4n^{-2}, \tag{34}$$

where $\overline{\gamma} = \lim_{N\to\infty}\left(\sum_{m=1}^{N}\frac{1}{m} - \log N\right) = 0.577\ldots$ is the Euler-Mascheroni constant.

Now, clearly for all $k = 1,\ldots,p^m-1$,

$$|1+\gamma_j+\gamma_j C(k,p^m,z_j)| \leq 1+\gamma_j+\gamma_j L(p^m).$$

Further,

$$\sum_{k=1}^{p^m-1}\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}}C(k,p^m,z_{s+1}) \leq \sum_{k=1}^{p^m-1}\frac{|T(k,p^m)|}{\phi(p^m)},$$

where

$$T(k,p^m) = \sum_{z\in\mathcal{Z}_{p,m}}\sum_{h\in B_{p^m}^*}\frac{\mathrm{e}^{2\pi\mathtt{i}kh_jz_j/p^m}}{|h|}.$$

Following the proof of [21, Theorem 5.10], we see that for $k = 1,\ldots,p^m-1$ we can write $T(k,p^m)$ as

$$T(k,p^m) = -2H(k,p^m) + V(k,p^m),$$

with

$$H(k,n) = \sum_{d|n}\nu\left(\frac{n}{d}\right)\gcd(d,k)\log\left(\frac{d}{\gcd(d,k)}\right)$$

and

$$V(k,n)=\sum_{d|n}\nu\left(\frac{n}{d}\right)\gcd(d,k)\varepsilon\left(\frac{n\gcd(d,k)}{d}\right),$$

where here ν denotes the well-known Möbius function from number theory and ε is the function introduced in (34). From [21] we see that for $k=1,\ldots,p^m-1$ the term $H(k,p^m)$ is non-negative, and indeed

$$\sum_{k=1}^{p^m-1}H(k,p^m)=\phi(p^m)\log p^m.$$

Further, as shown in [27], we can write

$$\begin{aligned}|V(k,p^m)|&\le\sum_{d|p^m}\left|\nu\left(\frac{p^m}{d}\right)\right|\gcd(d,k)\left|\varepsilon\left(\frac{p^m\gcd(d,k)}{d}\right)\right|\\&\le 4\sum_{d|p^m}\left(\frac{d}{p^m}\right)^2=4\sum_{d|p^m}\frac{1}{d^2}\le\frac{2\pi^2}{3}.\end{aligned}$$

Putting all this together we get

$$\begin{aligned}\sum_{k=1}^{p^m-1}\frac{|T(k,p^m)|}{\phi(p^m)}&\le 2\log p^m+\frac{2\pi^2(p^m-1)}{3\phi(p^m)}\\&\le 2\log p^m+\frac{2^{\kappa+1}\pi^2}{3},\end{aligned}$$

since $\frac{n}{\phi(n)}\le 2^\kappa$ for all n where κ is the number of distinct prime factors of n. Combining everything together and noting that for all positive integers n the inequality $L(n)\le 2\log n$ holds, we see that

$$\begin{aligned}&\frac{1}{\phi(p^m)}\sum_{z_{s+1}\in\mathcal{Z}_{p,m}}\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*,z_{s+1})\\&\le\frac{\gamma_{s+1}}{p^m}L(p^m)\prod_{j=1}^{s}(1+\gamma_j+\gamma_jL(p^m))\\&\quad+\frac{\gamma_{s+1}}{p^m}\left(2\log p^m+\frac{2^{\kappa+1}\pi^2}{3}\right)\prod_{j=1}^{s}(1+\gamma_j+\gamma_jL(p^m))\\&\le\frac{\gamma_{s+1}}{p^m}\left(4\log p^m+\frac{2^{\kappa+1}\pi^2}{3}\right)\prod_{j=1}^{s}\left(1+\gamma_j+\gamma_j\left(4\log p^m+\frac{2^{\kappa+1}\pi^2}{3}\right)\right)\\&=\overline{\Upsilon}_{p^m,s+1,\boldsymbol{\gamma}}.\end{aligned}$$

□

For $b\ge 1$ and $\boldsymbol{z}_s^*\in\mathcal{Z}_p^s$, define

$$\mathcal{B}_{p^m,s+1,\boldsymbol{\gamma}}(b;\boldsymbol{z}_s^*)=\left\{z_{s+1}\in\mathcal{Z}_p\,:\,\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*,z_{s+1})\le b\,\overline{\Upsilon}_{p^m,s+1,\boldsymbol{\gamma}}\right\}. \tag{35}$$

As before we again simply write μ for the measure μ_1. Then we have

Theorem 15. *Let p, m and s be positive integers. Then for any $b \geq 1$ we have*

$$\mu(\mathcal{B}_{p^m,s+1,\boldsymbol{\gamma}}(b\,;\boldsymbol{z}_s^*)) > 1 - b^{-1}.$$

Theorem 16. *Let p and s be positive integers and let $0 < m_1 \leq m_2$. Let $b_m \geq 1$ for $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} b_m^{-1} \leq 1$. Then there exists a $z_{s+1}^* \in \mathcal{Z}_p^s$ such that*

$$\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}((\boldsymbol{z}_s^*, z_{s+1})) \leq b_m \overline{\Upsilon}_{p^m,s+1,\boldsymbol{\gamma}} \quad \text{for} \quad m = m_1, \ldots, m_2.$$

Proof. The proof is analogous to that of Theorem 9. □

Theorem 17. *Let p, m and s be positive integers. Let $\boldsymbol{z}_s^*$ be chosen so that*

$$R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*) \leq b_m \overline{R}_{p^m,s,\boldsymbol{\gamma}}$$

and let z_{s+1}^* be chosen so that

$$\Upsilon_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_s^*, z_{s+1}^*) \leq b_m \overline{\Upsilon}_{p^m,s+1,\boldsymbol{\gamma}}.$$

Then

$$R_{p^m,s+1,\boldsymbol{\gamma}}(\boldsymbol{z}_{s+1}^*) \leq b_m \overline{R}_{p^m,s+1,\boldsymbol{\gamma}},$$

where $\boldsymbol{z}_{s+1}^* = (\boldsymbol{z}_s^*, z_{s+1}^*)$.

Theorem 18. *Let p and s be positive integers and let $0 < m_1 \leq m_2$. Let $b_m \geq 1$ for $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} b_m^{-1} \leq 1$. Then an algorithm equivalent to Algorithm 1 constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that for $m = m_1, \ldots, m_2$*

$$R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq b_m \overline{R}_{p^m,s,\boldsymbol{\gamma}}.$$

Finally, the fast CBC sieve construction may be used to find a generating vector $\boldsymbol{z}^*$ with small $R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}^*)$ for $m = m_1, \ldots, m_2$, again by making the necessary adjustments, i.e., replacing the worst-case error with the quantity $R_{p^m,s,\boldsymbol{\gamma}}$ and replacing the bound $E_{p^m,s,\alpha,\boldsymbol{\gamma}}$ with $\overline{R}_{p^m,s,\boldsymbol{\gamma}}$.

The following theorem now applies to generating vectors constructed using the fast CBC algorithm based on $R_{p^m,s,\boldsymbol{\gamma}}$.

Theorem 19. *Let p and s be positive integers and let $0 < m_1 \leq m_2$. Let $b_m \geq 1$ for $m = m_1, \ldots, m_2$ such that $\sum_{m=m_1}^{m_2} b_m^{-1} \leq 1$. Then an algorithm equivalent to Algorithm 2 constructs a vector $\boldsymbol{z}^* \in \mathcal{Z}_p^s$ such that for $m = m_1, \ldots, m_2$*

$$R_{p^m,s,\boldsymbol{\gamma}}(\boldsymbol{z}^*) \leq b_m \overline{R}_{p^m,s,\boldsymbol{\gamma}}.$$

7. Concluding remarks

Though we provide some useful constructions here, there are still some open questions. First let us address the meaning of "extensible". In the Introduction we wrote that the existence of good extensible lattice rules was shown in [13]. Here the meaning of extensible is from a practical point of view; namely that there exists a lattice rule whose generating vector $\boldsymbol{z}^*$ is such that one can obtain good lattice rules for all moduli $p, p^2, \ldots$. What would be an interesting result in this direction, but was not shown in [13], is the following:

For any generating vector of a good lattice rule in dimension s with a number of points p^m, there exists an extension of this generating vector such that one obtains a good lattice rule for some other number of points $p^{m'}$ with $m' \neq m$.

(Compare this statement with a probabilistic version in Remark 1. Further, an analogous result for the dimension is known if $s' > s$; see [3, 16].) Such a result would indeed be interesting, but at present it is not even known whether this statement is true, let alone how it can be made constructive. This seems to be a much more challenging question, as the probabilistic arguments used in [4, 13, 22] and here do not seem to apply; rather one would have to find some number theoretic reason to prove such a result (a constructive algorithm which achieves this might be even more difficult to obtain). Hence in terms of construction, what has been known until now is only the existence of good "embedded" lattice rules (embedded in the number of points n), i.e., lattice rules which work well for a whole range of moduli, rather than extensible lattice rules. Hence the algorithms introduced here are feasible constructions of good lattice rules, achieving what has been known until now about their existence. Thus everything known about the existence of extensible lattice rules has been made practical in this paper (see [4] for the analogue for polynomial lattice rules).

To make even more precise what we mean here, let us give an example of true extensibility. Namely, using the CBC algorithm, good lattice rules are truly extensible in the dimension. That is, if one is given a good extensible lattice rule in some finite dimension s, then one can add another coordinate to obtain a good lattice rule in $s+1$ dimensions [3, 16, 29]. On the other hand, such a lattice rule does not have to be embedded in the dimension: for example, construct a good Korobov lattice rule in dimension s (i.e., the generating vector is of the form $(1, z, z^2, \ldots, z^{s-1})$); then using the CBC algorithm we can add arbitrarily many coordinates to obtain a good lattice rule in $s' > s$ dimensions [7]. But until now we have not been able to prove that we can extract an $s-1$-dimensional good lattice rule from the s'-dimensional or s-dimensional lattice rule given at the beginning. Hence our lattice rule is extensible in the dimension, but not necessarily embedded (meaning that we can extract a good lattice rule from a given one in dimensions $s = 1, 2, 3, \ldots$). Using the CBC algorithm from dimension one onwards, we can of course obtain a lattice rule which is extensible and embedded in the dimension in this sense.

Thus, in this terminology, what was shown in [13] is the existence of a good lattice rule which is embedded in n and s simultaneously, and this has been made constructive in this paper. Note that in this paper we have even improved this result by showing the existence of a lattice rule which is embedded in n and extensible and embedded in s; this is also made constructive in our algorithms (which have been achieved by incorporating the CBC approach).

References

[1] N. Aronszajn, Theory of reproducing kernels. Trans. Amer. Math. Soc., 68 (1950), 337–404. MR0051437 (14:479c)

[2] R. Cools, F.Y. Kuo and D. Nuyens, Constructing embedded lattice rules for multivariate integration. SIAM J. Sci., 28 (2006), 2162–2188. Comput. MR2272256 (2007m:68324)

[3] J. Dick, On the convergence rate of the component-by-component construction of good lattice rules. J. Complexity, 20 (2004), 493–522. MR2068155 (2005h:65035)

[4] J. Dick, The construction of extensible polynomial lattice rules with small weighted star discrepancy. Math. Comp., 76 (2007), 2077–2085. MR2336283

[5] J. Dick, I.H. Sloan, X. Wang and H. Woźniakowski, Liberating the weights. J. Complexity, 20 (2004), 593–623. MR2086942 (2005h:65008)

[6] J. Dick, I.H. Sloan, X. Wang and H. Woźniakowski, Good lattice rules in weighted Korobov spaces with general weights. Numerische Mathematik, 103 (2006), 63–97. MR2207615 (2006m:65014)

[7] J. Dick and X. Wang, A hybrid construction method for good lattice rules in weighted Korobov spaces. Preprint.

[8] G.H. Hardy and E.M. Wright, An Introduction to the Theory of Numbers, 5th Ed., Clarendon Press, Oxford, 1979. MR568909 (81i:10002)

[9] F.J. Hickernell, A generalized discrepancy and quadrature error bound. Math. Comp., 67 (1998), 299–322. MR1433265 (98c:65032)

[10] F.J. Hickernell, My dream quadrature rule. J. Complexity, 19 (2003), 420–427. MR1984124 (2004e:41034)

[11] F.J. Hickernell and H.S. Hong, Computing multivariate normal probabilities using rank-1 lattice sequences. Scientific computing (Hong Kong, 1997), pp. 209–215, Springer, Singapore, 1997. MR1729661

[12] F.J. Hickernell, H.S. Hong, P. L'Ecuyer and C. Lemieux, Extensible lattice sequences for quasi-Monte Carlo quadrature. SIAM J. Sci. Comput., 22 (2000), 1117–1138. MR1785348 (2001h:65032)

[13] F.J. Hickernell and H. Niederreiter, The existence of good extensible rank-1 lattices. J. Complexity, 19 (2003), 286–300. MR1984115 (2004c:65015)

[14] F.J. Hickernell and H. Woźniakowski, Tractability of multivariate integration for periodic functions. J. Complexity, 17 (2001), 660–682. MR1881663 (2003g:65028)

[15] S. Joe, Construction of good rank-1 lattice rules based on the weighted star discrepancy. In: *Monte Carlo and Quasi-Monte Carlo Methods 2004.* (Niederreiter H. and Talay D., eds.), pp. 181–196, Springer, Berlin, Heidelberg, New York, 2006. MR2208709 (2006m:65048)

[16] F.Y. Kuo, Component-by-component constructions achieve the optimal rate of convergence for multivariate integration in weighted Korobov and Sobolev spaces. J. Complexity, 19 (2003), 301–320. MR1984116 (2004c:41066)

[17] F.Y. Kuo and S. Joe, Component-by-component construction of good lattice rules with composite number of points. J. Complexity, 18 (2002), 943–976. MR1933697 (2003j:65016)

[18] G. Larcher, On the distribution of an analog to classical Kronecker-sequences. J. Number Theory, 52 (1995), 198–215. MR1336745 (96g:11098)

[19] G. Larcher and H. Niederreiter, Generalized (t, s)-sequences, Kronecker-type sequences, and Diophantine approximations of formal Laurent series. Trans. Amer. Math. Soc., 347 (1995), 2051–2073. MR1290724 (95i:11086)

[20] H. Niederreiter, Quasi-Monte Carlo methods and pseudo-random numbers. Bull. Amer. Math. Soc., 84 (1978), 957–1041. MR508447 (80d:65016)

[21] H. Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods, CBMS–NSF Series in Applied Mathematics 63, SIAM, Philadelphia, 1992. MR1172997 (93h:65008)

[22] H. Niederreiter, The existence of good extensible polynomial lattice rules. Monatsh. Math., 139 (2003), 295–307. MR2001711 (2004j:11087)

[23] H. Niederreiter, Constructions of (t, m, s)-nets and (t, s)-sequences. Finite Fields and their Applications, 11 (2005), 578–600. MR2158777 (2006c:11090)

[24] D. Nuyens and R. Cools, Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces. Math. Comp., 75 (2006), 903–920. MR2196999 (2007a:65032)

[25] D. Nuyens and R. Cools, Fast construction of rank-1 lattice rules with a non-prime number of points. J. Complexity, 22 (2006), 4–28. MR2198499 (2006k:65063)

[26] I.F. Sharygin, A lower estimate for the error of quadrature formulas for certain classes of functions. Zh. Vychisl. Mat. i Mat. Fiz., 3 (1963), 370–376. MR0150952 (27:938)

[27] V. Sinescu and S. Joe, Good lattice rules with a composite number of points based on the product weighted star discrepancy. In Monte Carlo and Quasi-Monte Carlo Methods 2006 (A. Keller, S. Heinrich and H. Niederreiter, eds.), Springer-Verlag, Berlin, Heidelberg, pp. 645–658.

[28] I.H. Sloan, S. Joe, Lattice Methods for Multiple Integration, Oxford University Press, Oxford, 1994. MR1442955 (98a:65026)

[29] I.H. Sloan, F.Y. Kuo and S. Joe, Constructing randomly shifted lattice rules in weighted Sobolev spaces. SIAM J. Numer. Anal., 40 (2002), 1650–1665. MR1950616 (2003m:65031)

[30] I.H. Sloan and H. Woźniakowski, When are quasi-Monte Carlo algorithms efficient for high-dimensional integrals? J. Complexity, 14 (1998), 1–33. MR1617765 (99d:65384)

[31] X. Wang, I.H. Sloan and J. Dick, On Korobov lattice rules in weighted spaces. SIAM J. Numer. Anal., 42 (2004), 1760–1779. MR2114300 (2005j:65006)

School of Mathematics and Statistics, University of New South Wales, Sydney 2052, Australia

E-mail address: josi@maths.unsw.edu.au

Institut für Finanzmathematik, Universität Linz, Altenbergstrasse 69, A-4040 Linz, Austria

E-mail address: friedrich.pillichshammer@jku.at

School of Mathematics and Statistics, University of New South Wales, Sydney 2052, Australia

E-mail address: benjw@maths.unsw.edu.au